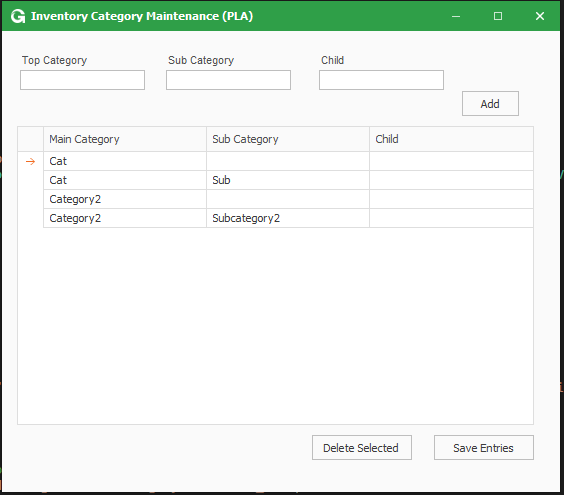
**Technical aspects of the web integration for Paragon with WooCommerce through Pantheon**

We created an integration to WooCommerce for Paragon to allow Paragon’s clientele to make orders through the website instead of having to call in and have the front desk create their orders.

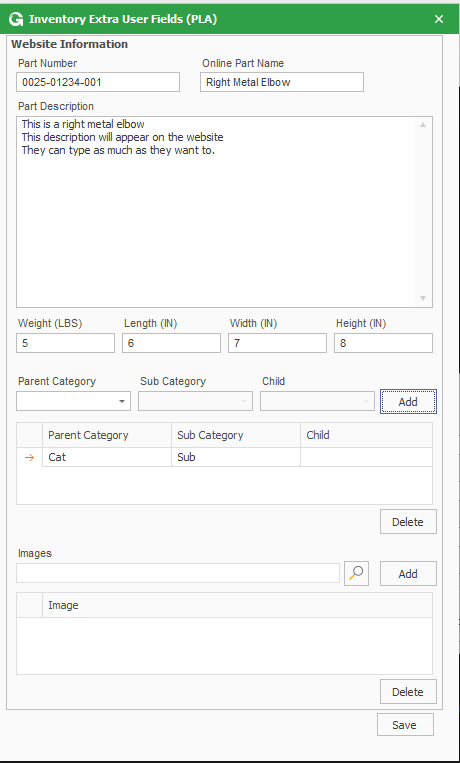
This integration uses REST API to communicate with WooCommerce. It interacts in two different sections.

Inventory information is first created in Global Shop. There are specific informational requirements that WooCommerce needs before a part can be added on the website, including assigning categories to them.

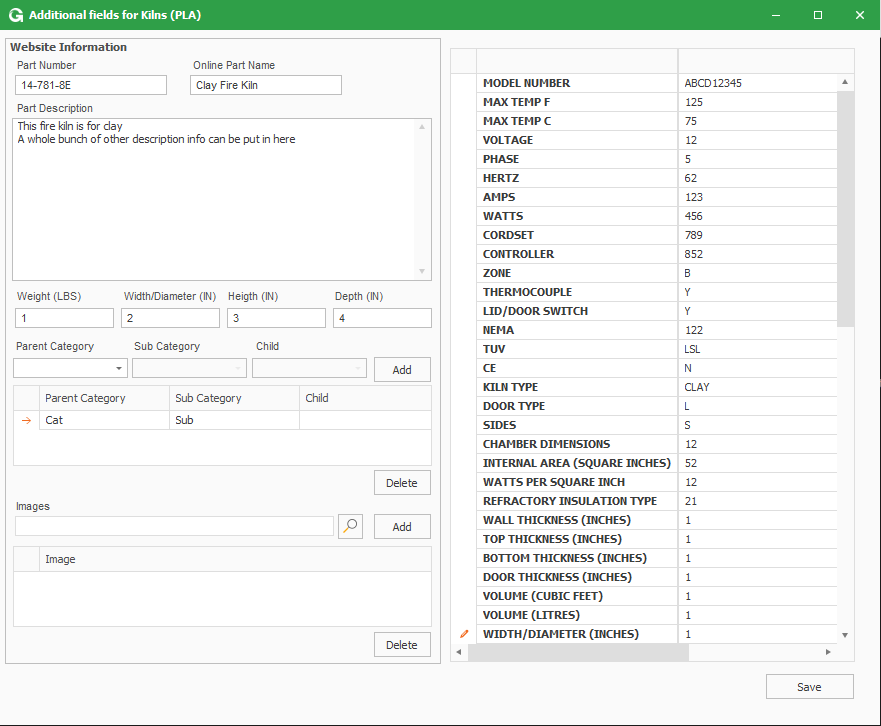
There is a custom menu item in Inventory > Administration > Category Maintenance where users can create categories that can be used to label inventory parts. Categories are created on the website directly from this screen, and it is required that any categories that will be used on parts from Global Shop be added on this screen.



The Paragon users label the parts that they want available to sell on their website by typing YES on the “Website Part” textbox in Inventory Master. Inventory information that will be displayed on the website can be filled into the custom screen that can be accessed through the “Website Info Part” (Script 3) button in Inventory master.



Paragon manufactures kilns and any parts classified as kilns must have additional information filled in. Inventory parts which are kilns and will be uploaded into the website are required to have the “Kiln Part” textbox filled in with YES. The additional attributes for kilns can be added in Global Shop through the Website Info Kiln (Script 2) button on Inventory Master.



The inventory information gets uploaded into the website through a script that is set to run unattended every 15 minutes. This script will check to see if any inventory has been added or modified and will send updates into the website. There is also a custom menu item set up for them to use that will flag all online parts as modified to completely re-upload everything. This was done due to the way that they update pricing.

The project will also process information coming from the website, including new customer information and new orders. This is done through a script that runs every 15 minutes. The script will verify that the order has not yet been created in GSS by comparing the web ID number to the Customer PO numbers in the orders in GSS. If the order needs to be created, the script will first verify that the customer account exists in GSS as well. If the customer account does not exist, the script will create the customer account in GSS using the information provided by the order from the Website. Inventory in the order lines is verified first by reading in the web id of the part and checking it against the inventory that has been flagged as online in GSS. If the inventory is not found, the script will check to see if the inventory part has a meta-data set up for ‘gss\_part’. If so, it will read that value in and verify that part number exist.

The current iteration of the process ends after the order is created in GSS. Every process post order creation is handled manually by the Paragon users.

Technical aspects and common issues/troubleshooting

The original project was quoted to be an addition to an original UF project to make it cheaper for the customer, but they decided not to use the original piece. As a consequence, the structure of the custom database tables is messy, and a bunch of the code in the script is not used. If the script with the user fields needs to be modified, it is suggested that the user run through the process with a debugger so they can understand the flow of the procedure.

The original project was quoted to integrate with a website hosted by cazbah, so if a script is ever seen using the cazbah website, it is most likely an outdated version. The final version of the project uses pantheonsite: https://live-pishop.pantheonsite.io/wp-json/wc/v3/. The connection is made through REST API, using basic authorization. The credentials are hard coded in the script, and the same credentials can be used to connect into their dashboard on their website if anything needs to be verified.

The inventory updates rely entirely on the web id that the project records upon the item creation on the website. If the part gets created directly on the website, the project will not be able to update the information on the website, and it will not be able to create orders in GSS for that part.

The program uses libraries to create customers and orders. The order upload library is a custom version that works with this program itself.

There is no current method implemented on the programs to handle the deletion of orders/inventory/customers. If any deletion needs to be done, they have to do everything manually. If they delete a part from the website and it is not removed from the inventory parts, this will cause the upload to crash. If this happens, the program will flag the inventory part as updated so that when the upload runs again, it will resume in the next part.

The images portion of the project was scrapped, so there is code in there for that, but it is not functional and is mostly commented out.